4-W

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/894,663	06/28/2001	N. David Naegle	5181-56601	6969

7590 01/24/2005

EXAMINER
WOODS, ERIC V

ART UNIT	PAPER NUMBER
2672	

Jeffrey C. Hood
MEYERTONS, HOOD, KIVLIN, KOWERT & GOETZEL
P.O. Box 398
Austin, TX 78767

DATE MAILED: 01/24/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/894,663	NAEGLE ET AL.	
	Examiner	Art Unit	
	Eric V Woods	2672	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *10 June 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-30* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-30* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *28 June 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Response to Arguments

1. Applicant's arguments – see pages 12-15, filed 04 June 2004 (termed 'Response' for reference within this action) -- with respect to the rejection(s)of claim(s) 1-30 under 35 U.S.C. 102(e) have been fully considered and are persuasive. Therefore, the rejections have been withdrawn. However, upon further consideration, a new ground(s) of rejection is made in view of Spiegel et al (US 5,615,282) in view of Tsay (US 5,673,215) and Harris (cited below).

2. Examiner points out that the 'winner-take-all' terminology used by applicant does not in fact specify anything at all; that is, winner-take-all circuits and methods are known in the art, but they take far different forms than that shown by applicant. The words 'winner-take-all' could be removed from all claims and the claims scope and the metes and bounds of the claims would not be (functionally) altered. As such, any structure that performs the recited functionality will be used, and the wording, unless it adds specific scope to the claim, will not be treated as an additional (functional) limitation.

3. Further responses to claim-specific arguments are covered in the new grounds for rejection for the claims set forth below.

Claim Rejections - 35 USC § 112

4. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

5. Claim 3 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

6. Claim 3 recites the limitation "the WTA output bit" in line 2. There is insufficient antecedent basis for this limitation in the claim. Applicant may intend on claim 3 to depend on claim 2, as there is a "WTA output bit" in that claim.

7. Where applicant acts as his or her own lexicographer to specifically define a term of a claim contrary to its ordinary meaning, the written description must clearly redefine the claim term and set forth the uncommon definition so as to put one reasonably skilled in the art on notice that the applicant intended to so redefine that claim term. *Process Control Corp. v. HydReclaim Corp.*, 190 F.3d 1350, 1357, 52 USPQ2d 1029, 1033 (Fed. Cir. 1999). The term "winner-take-all" bit in claims 1-30 is used by the claim to mean "select bit" or "control bit", while the accepted meaning for "winner-take-all" in the digital electronic art is "circuits that produce 1 when their input falls within a certain set of parameters, e.g. the input x_l is among the k-th largest inputs." The term is indefinite because the specification does not clearly redefine the term.

8. Claims 1-30 rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. That is, there is no such thing as a winner-take-all bit. Applicant is using the term "winner-take-all bit" in place of the terms "select bit" or "control bit" and as such it renders the claim indefinite, as the term has no positive

benefit in the claim, adds no structural or functional limitations, and renders the claim indefinite as it is confusing and again has no purpose.

9. Claim 27 is also rejected as indefinite because of the terminology "plurality of samples in response to said graphics data" since it is unknown what the plurality of samples are – e.g. pixels, portions of a picture, regions, et cetera.

Claim Rejections - 35 USC § 103

10. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

11. Claims 1, 10, and 23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Spiegel in view of Tsay.

12. As to claim 1,

An adder tree for adding numbers comprising:

-One or more addition levels including a top addition level and a bottom addition level, wherein a summation of said numbers begins at said top level and propagates through said one or more addition levels, wherein each of said addition levels comprises one or more adder cells; (Spiegel Fig. 14 and referenced 'Adder Tree' in Figs. 10-12)

-Wherein each of said adder cells is configured to receive a first input operand, a second input operand, a first winner-take-all (WTA) bit and a second WTA bit, and to generate a first output operand, wherein the first output operand equals the first input operand if the first WTA bit is high, wherein the first output operand equals to the

second input operand if the second WTA bit is high; and (Spiegel Fig. 14, each adder cell is shown to receive two input operands; Spiegel Fig. 12, there is a selector unit shown as element 202, which clearly selects a number for input before it is passed into the adder tree.)(Tsay clearly shows in Fig. 3 the use of multipliers before each adder, which clearly matches applicant's structure shown in applicant's drawing Fig. 33A.)

-Wherein each of said one or more adders at the top addition level receives two of said numbers as the corresponding first input operand and the second input operand. (Spiegel Fig. 14 and referenced 'Adder Tree' in Figs. 10-12)

Firstly, Spiegel in Fig. 14 as well as Figs. 10-12 shows an adder tree that clearly has "one or more additional levels" that receive "two of said numbers as corresponding first and second operant" as recited in claim 1. Applicant does not dispute this conclusion (Response, pgs 12-13), but rather directs all the response to the fact that Spiegel does not apparently show the claimed 'winner-take-all' bits, and Spiegel does not expressly teach that limitation. Spiegel does however teach the use of a select section before the adders, which could obviously be multiplexers (e.g. element 202 in Fig. 12).

Reference Tsay is directed to the same problem solving area, as it is directed to the use of adders, which are notoriously generic pieces of digital logic. Reference Tsay as shown above uses two multiplexers before an adder. Clearly, it would have been obvious to modify the structure of Tsay to include zeros as one of the inputs to the multipliers to create an adder cell that had the recited 'winner-take-all' structure such that the multiplexers could be used to avoid passing data values and instead pass in a

zero when the multiplexer was activated. The point of using zeros is so that the other input of the multiplexer is not floating and is a fundamental of the electrical engineering art. Please note that clearly, there is an annotation on Fig. 3 of Tsay that states that the elements on the right are an "Adder Block for Divider", which would clearly allow the inclusion of such components in the adder cell.

It would have been obvious to one having ordinary skill in the art at the time the invention was made to combine the Spiegel adder tree and the adder structure of Tsay, as this would modification would allow the user or system that controlled an adder block within the tree of Spiegel to specify whether or not a branch of the adder tree should be ignored (e.g. passed a zero through the multiplexer The point of using zeros is so that the other input of the multiplexer is not floating and is a fundamental of the electrical engineering art.) or not.

13. As to claim 10,

The adder tree of claim 1, wherein the first output operand equals zero, the first input operand, the second input operand, or the sum of the first input operand and the second input operand if the first and second WTA bits are low.

Reference Spiegel does not expressly teach this limitation, but reference Tsay does. With the addition of the multiplexers, any of the four combinations would be possible based on which control signals were sent along the winner-take-all / select control lines. As recited in claim 1, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the multiplexers so that one input on each was tied to zero. In any case, the addition of the multiplexers from Tsay means

that the combined circuit inherently produces one of the four results specified above when the control signals are both low. The point of using zeros is so that the other input of the multiplexer is not floating and is a fundamental of the electrical engineering art.

It would have been obvious to one having ordinary skill in the art at the time the invention was made to combine the Spiegel adder tree and the adder structure of Tsay, as this would modification would allow the user or system that controlled an adder block within the tree of Spiegel to specify whether or not a branch of the adder tree should be ignored (e.g. passed a zero through the multiplexer)(The point of using zeros is so that the other input of the multiplexer is not floating and is a fundamental of the electrical engineering art) or not.

14. As to claim 23,

A method comprising:

(a) Receiving a plurality of samples, which represent at least a portion of a graphical image; (color CRT display 33:15-25; Fig. 42/1 and 42/2 showing a rendering unit attached to the recited Microwhisper workstation; see 81:55-67 and 82:1-26 for details on that computer and how the rendering unit is connected and the results of that processing; since the recited workstation is based on an Intel 386 type workstation, it is inherent that it used an ISA or PCI type bus that are known to one of ordinary skill in the art to be able to access main memory. Further, in 6:1-30 it is disclosed that the graphics data is stored in region-by-region format, which would clearly mean that the rendering unit would be operable to render a plurality of samples (i.e. pixels or regions, whichever interpretation is taken, thusly teaching image processing by portion).

(b) Computing a plurality of numeric values in response to the plurality of samples (relevance values are assigned – at least three levels, see Spiegel 11:5-16)(further, color coordinate values for each CT/LW pixel or sub-group are clearly calculated; these clearly represent a "plurality of numeric values") (Spiegel 11:15-30, where the image is partitioned into sub-groups, which clearly would include numeric values that were computed as respect to above, given that such values are associated with each pixel, as well as 21:5-35 – namely, color coordinate values and relevance values as numeric data)(data valid (DV) signals are computed for each pixel as well – see for example Spiegel 2:41-55);

(c) Determining a plurality of winner-take-all (WTA) bits based on positions of said samples with respect to a pixel center, wherein each of said WTA bits corresponds to one of said samples; (Spiegel 10:50-67 and 11:1-15, particularly 11:1-5 for distance from or position with respect to a pixel center of a sub-portion.)(The WTA bits could be either the relevance values or the data valid signals for each pixel, and it would be obvious to compute such with respect to the pixel center of the sub region of Spiegel as set forth in 10:50-67 and 11:1-5)

(d) Forming groups comprising two or more of said numeric values and two or more of said WTA bits; (Spiegel 11:15-30, where the image is partitioned into sub-groups, which clearly would include numeric values that were computed as respect to above, given that such values are associated with each pixel, as well as 21:5-35 – namely, color coordinate values and relevance values as numeric data)(obviously, such relevance

values and data valid (DV) signals would accompany the signals during the division into such groups as set forth immediately above)

(e) Generating a plurality of intermediate values and a corresponding plurality of intermediate WTA bits, wherein each of said intermediate values and the corresponding intermediate WTA bit correspond to one of said groups; (clearly, the adder trees of Spiegel teach the idea of a plurality of intermediate bits, because after processing the subgroups are sent into the adder tree, which converges to a final value, which is clearly the output pixel.)(Reference Tsay teaches the modification of the adder cells to have multiplexer such that the adder cells support control bits as set forth in the rejection to claim 1, the relevant portions of which are hereby incorporated by reference in their entirety)(Therefore, the data valid bits or the relevance levels could be the control / select bits for the multiplexers, which would fulfill the requirement of intermediate WTA bits, as they would logically propagate through the adder tree along with their values.)(Further, it would obvious to send signals into the adder tree in the order that they are referred to above).

Wherein each of said intermediate values equals:

-A summation of said numeric values in the corresponding group in response to none of said WTA bits of the corresponding group being set; or (clearly, if none of the WTA bits were asserted, the adders would function normally, that is, both of the input operands would be summed and sent out the other sides – this is *prima facie* obvious)

-One of said numeric values in the corresponding group in response to one of said WTA bits of the corresponding group being set; (Also as set forth above and in the

rejection to claim 1, it would be obvious to have tied one of the inputs to the multiplexers to zero such that a select bit (what applicant calls WTA bits) would thusly send a zero to the adder, avoiding further unnecessary propagation of the target value through the adder tree)(The point of using zeros is so that the other input of the multiplexer is not floating and is a fundamental of the electrical engineering art.)

-Wherein each of said intermediate WTA bits is generated by ORing the WTA bits of the corresponding group; and (It would be obvious to OR the inputs as the values propagated down the column, as the multiplexers in the adder cells further down the adder tree would require inputs, which would logically be provided by the values from the top level(s) of the tree, since a value that was valid or was asserted should continue to be propagated through the tree structure, and obviously if that value were valid the next cell should know. However, using an AND gate for this purpose would be pointless; for example, if two valid bits arrived at an adder cell, then they would have to be added in a normal fashion, as they had the same level of priority. Thusly, the output of such a cell would not have preempting priority because it had been added. Only when one value had priority or validity should the priority / WTA bit continue to be propagated to further adders (e.g. to the select values of the multiplexers in the combined adder cell below it. Thusly, given that the goal would be to use the simplest possible logic and that a NOR gate would produce the same results as an AND gate, the only logic gate left would an OR gate.)

-Repeating (d) and (e) until a single resultant value is obtained, wherein each of the groups in a second or succeeding iteration of (d) comprises two or more of the

intermediate values from a previous iteration of (e). (The adder tree obviously repeats itself and iterates downwards until it results in a single resultant value. *Prima facie*, when using an adder tree each second or succeeding group would utilize intermediate values propagating down the tree.)

Therefore, for the reasons set forth above, the Spiegel reference does teach all of the limitations except explicitly setting forth the structure of the adder cell, which is taken from Tsay as set forth in the rejection to claim 1, which is herein incorporated by reference. The logic behind modifying the Spiegel / Tsay references is provided in point by point fashion next to the steps set forth in the method claim above. The references are analogous art, et cetera (see claim 1 for that justification). As such, It would have been obvious to one having ordinary skill in the art at the time the invention was made to combine the references of Spiegel and Tsay and so modify the combination as set forth above, because the above method is no more than a broadly written claim that reads on adder trees and methods as used by the Spiegel reference.

It would have been obvious to one having ordinary skill in the art at the time the invention was made to combine the Spiegel adder tree and the adder structure of Tsay, as this would modification would allow the user or system that controlled an adder block within the tree of Spiegel to specify whether or not a branch of the adder tree should be ignored (e.g. passed a zero through the multiplexer) or not, which would clearly allow the avoiding of unnecessary operations by the adder cells.

15. Claim 9 is rejected under 35 U.S.C. 103(a) as unpatentable over Spiegel in view of Tsay as applied to claim 1 above, and further in view of Harris et al (US 4,021,654)('Harris').

As to claim 9,

The adder tree of claim 1 further comprising buffer registers interposed between a first addition level and a second addition level of the adder tree to temporarily store output operands generated by adder cells of the first addition level prior to their presentation to the second addition level.

- Reference Harris is analogous art – as it clearly teaches digital filtering.

References Spiegel and Tsay do not explicitly teach this limitation. Reference Harris teaches use of asynchronous input buffers (Fig. 3, element 100 has an asynchronous buffered input (5:10-25)). Clearly, if the adder tree were asynchronous, e.g. if the multipliers were not synchronous in their rate of return, buffers would be necessary such that inputs into the adder tree were not lost, and reference Harris clearly teaches the use of asynchronous buffers. See Spiegel col. 54, lines 34-54 where clearly Spiegel, by introducing multiple pipeline registers that introduce suitable delays so that all values can propagate through the logic tree simultaneously. It would have been obvious to one having ordinary skill in the art at the time the invention was made to add buffers to every adder to accommodate the use of asynchronous logic and allow for differing times of return. It would have been obvious to one having ordinary skill in the art at the time the invention was made to combine the convolving systems of Spiegel and Tsay with the asynchronous buffers of Harris for the reasons set forth

above and to allow more flexibility in designing the circuit, and that Spiegel clearly teaches the use of delay buffers to allow operands to reach the adder tree before propagating to the next level.

16. Claims 15-17 and 27-30 are rejected under 35 U.S.C. 103(a) as unpatentable in view of Spiegel.

17. As to claim 15,

A pixel computation unit comprising:

-A sample request unit configured to read samples from a sample buffer, and select one or more of the samples residing within one or more filter regions; (Spiegel Fig. 8, the convolution scaling device 96 in element 94 obtains its information from CT row buffers 86, which clearly (as set forth below) pull data from the row buffers as needed for the N x N convolution window)(clearly the preferred embodiment also operates by regions anyway – see 6:10-26)

-One or more multiplication units configured to multiply a first sample component of each selected sample by a corresponding coefficient to generate one or more weighted first sample components; (Spiegel Fig. 12, element 218)

-A first adder tree configured to receive the one or more coefficients used to obtain the one or more weighted first sample components, and to generate a coefficient sum comprising a sum of the one or more coefficients, wherein the first adder tree is further configured to receive the one or more weighted first sample components from the one or more multiplication units, and to generate a first summation of the weighted first sample components; and (Spiegel Fig. 12, element 220)

-A division unit configured to divide the first summation by the coefficient sum to obtain a first pixel value. (Spiegel 56:22-32, which clearly states that the selector divides the sum of the valid CT pixel color values by the sum of validity bits, which would clearly yield the appropriate pixel value).

Applicant acknowledges that the Spiegel reference has the multiplier and division units as well as a first adder tree by not disputing these points. Applicant's arguments to this effect are limited to two assertions:

a) That Spiegel does not teach or suggest an adder tree configured to "receive the one or more coefficients used to obtain ... components and to generate a coefficient sum ..."

b) That Spiegel does not teach or suggest a "sample request unit" that can "select one or more of the samples residing within one or more filter regions."

In Response, page 14, applicant recites Spiegel's description of a 5x5 convolution (col. 46, lines 50-57). The relevant portion states "...the 5x5 convolution computation on the first "window " of CT pixels defined in the image p_{ij}, namely the first five pixels of the first five rows..." Digital convolution is *prima facie* a filtering operation. Filtering operations are illustrated in 67:30-55, where it is shown that a filtering operation is implemented using a matrix with weighting coefficients, *which is exactly the operation that the convolution operation performs.* Clearly, the window of chosen pixels is a "filter region". Secondly, the convolution operation is performed while obtaining data from row buffers (element 86, Fig. 8, Spiegel), such that obviously the convolution scaler (elements 94 / 96, Fig. 8, Spiegel) is operative to "select one or more of the

samples residing within one or more filter regions", where clearly the filter region is the convolution window and the selected portion is the relevant portion of a row drawn from the row buffers.

In Spiegel, first of all, the coefficients used for the convolution operation are used in matrix form. The digital convolution operation fundamentally represents a multiplication of matrices. An example can be found in Spiegel 67:30-55, where two examples of convolution operations and matrices are given. Further, in Spiegel 13:1-35, clearly it is stated that the method "includes the steps of providing a first plurality of first coefficients corresponding in number to a first plurality of locations ... providing, for each position along the second dimension of the array, a weighted value sum including a weighted sum of the values along the first dimension..." Then, clearly, in Spiegel 67:59-67 and 68:1-5, it is taught "...it is advantageous to provide matrix coefficients which sum to 1 over each row..." Clearly, if the sum of the coefficients across a row is one, there is no need to divide by the sum of the coefficients to normalize, as division by one changes nothing. After all, the concept of normalizing is quite well known, and Spiegel teaches normalization in 56:23-33 for example, where pixels are divided by a sum.

Secondly, the recited adder tree can clearly be used to add or sum any desired set of numbers. Thusly, an adder tree capable of summing the weighted values after multiplication (see 13:10-36, that is, after the convolution operation, which is inherently multiplication) is taught. Obviously, such an adder system could easily be set up to the sum the weighting coefficients themselves, rather than the weighted results of the

convolution. The adder tree as configured in Spiegel is set up to receive the sum of the values after multiplication as shown in Figure 12.

Further, it is inherent in digital convolution that a digital filter (finite impulse response (FIR) or infinite impulse response (IIR)) -- when a delta function $\delta(z)$ (a digital approximation of a Dirac delta function) (assuming z-transform domain) is convolved with a system having an output $y(z)$ and a system response $h(z)$, according to the equation $x(z)*h(z)=y(z)$ where $x(z)$ is the input and * represents the convolution operator, that $\delta(z) * h(z) = h(z)$. That is, a delta function will cause the system to output the system response function $h(z)$, which in this case would be the weighting coefficients of the digital filter in question. Clearly, the adder tree generates the sum of the weighted values after they pass through the multipliers. If a delta function were applied to the multipliers (in this case, very simple to do, simply an identity matrix having the size of the convolution window), the row coefficients would be sent out of the multipliers, which would then cause the adder tree to inherently produce sums of weighted coefficients.

So clearly, the system of Spiegel can be utilized to generate the recited "sum of one or more coefficients". Further, Spiegel clearly teaches above that in a preferred embodiment that the sum of the coefficients would be one, thus negating the need to normalize the summed values by dividing by the sum of the coefficients. If the sum of the coefficients were not one, then it would be appropriate to sum the coefficients and divide the sum of weighted values by the sum of the coefficients to normalize the sum. However, as shown in Spiegel Fig. 8, the selector 90 receives the data from the

convolution scaler, which contains the adder tree. Spiegel 56:22-32, which clearly states that the selector divides the sum of the valid CT pixel color values by the sum of validity bits, which would clearly yield the appropriate pixel value, which is clearly a mathematical normalization operation, which is defined by the American Heritage College Dictionary as "to make normal, esp. to conform to a standard or norm".

Finally, Spiegel performs normalization in several places, for example 61:60-67 and 62:1-8.

Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to modify the division unit in the selector of Spiegel to perform normalization of the sums of weighted values after leaving the convolution scaler by firstly generating the sum of weighting coefficients by passing an N x N (with N x N being the dimensions of the convolution window, for example 5 x 5 as cited by applicant on Response page 14) identity matrix through the convolution scaler (namely, the adder tree) and then dividing the sum of weighted values by the sum of weighting coefficients.

Thusly, both of applicant's arguments have been answered concerning claim 15 and it stands rejected as set forth above.

18. As to claim 16,

The pixel computation unit of claim 15, wherein the one or more multiplication units are further configured to multiply a second sample component of each selected sample by the corresponding coefficient, and thus, to generate one or more weighted second sample components; (see Spiegel 13:10-45, where it is disclosed that the system /

method performs convolution in multiple dimensions and has multiple sets of coefficients that multiply the second sample component or dimension by the second set of weighting coefficients, and clearly element 94/96 is a pixel computation unit; Fig. 12 shows the multiplier unit as element 218)

-Wherein the first adder tree is further configured to receive the one or more weighted second sample components from the one or more multiplication units, and to generate a second summation of the weighted second sample components; (the adder tree is in element 94/96 of Fig. 8 and is shown in detail in Figs. 10-12, particularly Fig. 12, element 220, and it receives the weighted component values and sums them, as it is clearly shown to receive the data from the multipliers 218, and *prima facie* generates sums as the multipliers get data from the overhead M-dimensional matrix 190 and multiply them by the coefficients from the coefficient table 216)

-Wherein the division unit is further configured to divide the second summation by the coefficient sum to obtain a second pixel value. (The division unit is in element 90 the selector as previously established above in the rejection to claim 15) (Spiegel 56:22-32, which clearly states that the selector divides the sum of the valid CT pixel color values by the sum of validity bits, which would clearly yield the appropriate pixel value).

Reference Spiegel teaches all of the limitations except expressly stating that the division unit is configured to divide the second summation by the coefficient sum. See the above logic in the rejection to claim 15 for an explanation of why it would be obvious to modify Spiegel to perform normalization via sum of weighting coefficients if the sum were not one. Obviously, since the preferred embodiment convolves N-dimensional

data where, for example, N is at least equal to three, see Spiegel 8:50-56. If it would be obvious to normalize data in one dimension, it would be obvious to modify the selector to normalize for all N dimensions.

Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to modify the division unit in the selector of Spiegel to perform normalization of the sums of weighted values after leaving the convolution scaler by firstly generating the sum of weighting coefficients by passing an N x N (with N x N being the dimensions of the convolution window, for example 5 x 5 as cited by applicant on Response page 14) identity matrix through the convolution scaler (namely, the adder tree) and then dividing the sum of weighted values by the sum of weighting coefficients.

19. As to claim 17,

The pixel computation unit of claim 15 further comprising a second adder tree, wherein the one or more multiplication units are further configured to multiply a second sample component of each selected sample by the corresponding coefficient, and thus, to generate one or more weighted second sample components; (second adder tree, Spiegel col. 54, lines 34-54, where it is disclosed that m adder trees can be used, which clearly correspond to the dimensionality of the images or data sets being convolved. As discussed in the above rejection to claim 15, Spiegel convolves multi-dimensional data and performs multiple sets of multiplications)

-Wherein the second adder tree is further configured to receive the one or more weighted second sample components from the one or more multiplication units, and to

generate a second summation of the weighted second sample components; (the adder tree is in element 94/96 of Fig. 8 and is shown in detail in Figs. 10-12, particularly Fig. 12, element 220, and it receives the weighted component values and sums them, as it is clearly shown to receive the data from the multipliers 218, and *prima facie* generates sums as the multipliers get data from the overhead M-dimensional matrix 190 and multiply them by the coefficients from the coefficient table 216. As such, and from the above stated Spiegel location in col. 54, lines 34-54, it would be obvious that the second adder tree does in fact perform the above recited functionality, and it would be notoriously obvious to modify it to do so if it did not already do so, as having multiple adder trees and not using them to compute different things would be pointless, particularly given that multiple dimensional multiplications are being carried out)

-Wherein the first adder tree and the second adder tree respectively compute the first summation and the second summation in parallel. (Again, see Spiegel col. 54, lines 34-54 where clearly Spiegel, by introducing multiple pipeline registers that introduce suitable delays so that all values can propagate through the logic tree simultaneously).

Reference Spiegel teaches all of the limitations except expressly stating that the second adder tree is configured to perform its operation in parallel. Obviously, since the preferred embodiment convolves N-dimensional data where, for example, N is at least equal to three, see Spiegel 8:50-56. It would have been obvious to modify the system of Spiegel to operate the adder trees in parallel, as clearly Spiegel convolves in multiple dimensions as set forth above, and Spiegel further teaches using delay circuits to cause the values to all propagate through the adder tree simultaneously thusly providing

support to using all the adder trees simultaneously (e.g. buffering them until all values arrive, then operating all in parallel to get results). Also, the very nature of an adder tree is parallel operation, so it would be obvious to extend that to multiple, simultaneously operating adder trees.

Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to modify the multiple adder trees of Spiegel to operate in parallel as set forth above by buffering the results of multiplication (with N x N being the dimensions of the convolution window, for example 5 x 5 as cited by applicant on Response page 14 and the multiple adders cited by Response page 13), and then performing operations in parallel.

20. As to claim 27,

A computer system comprising: (Spiegel 35:13-32)

-A central processing unit (CPU); (Spiegel 35:13-32)

-A main system memory coupled to said CPU; and (inherent to personal computer / workstation recited in 35:13-32; software requires memory to be executed)

-A graphics system comprising: (obviously, if the invention can be executed in software, then it *prima facie* requires a graphics system)

-A rendering unit operable to receive graphics data from said main system memory, wherein said rendering unit is operable to render a plurality of samples in response to said graphics data; (color CRT display 33:15-25; Fig. 42/1 and 42/2 showing a rendering unit attached to the recited Microwhisper workstation; see 81:55-67 and 82:1-26 for details on that computer and how the rendering unit is connected and the results

of that processing; since the recited workstation is based on an Intel 386 type workstation, it is inherent that it used an ISA or PCI type bus that are known to one of ordinary skill in the art to be able to access main memory. Further, in 6:1-30 it is disclosed that the graphics data is stored in region-by-region format, which would clearly mean that the rendering unit would be operable to render a plurality of samples (i.e. pixels or regions, whichever interpretation is taken).

-A buffer coupled to said rendering unit, wherein said buffer is operable to store said plurality of samples; and (Spiegel 6:35-40 and 35:6-12 clearly show an embodiment that contains multiple buffers for images to be stored)

-A sample-to-pixel calculation unit coupled to said buffer, wherein said sample-to-pixel calculation unit is operable to generate a plurality of pixels, wherein said sample-to-pixel calculation unit comprises a first adder tree; (Clearly the system must convert samples to pixels, since it inherently outputs the data to a display – see Fig. 42/2 where the Screen VLSI is shown, which is output to a monitor (this is also inherent to a personal computer workstation designed to process video or images – clearly they must be made visible and as stated above, output means are listed to include color CRT systems))

-Wherein the first adder tree comprises a plurality of adder cells coupled in a tree configuration, wherein each adder cell is configured to receive two or more input values and to generate an output value, wherein the output value equals one of said input values if said adder cell receives an asserted winner-take-all signal corresponding to said one of the input values, wherein the output value equals a sum of the input values or any subset thereof if said adder cell receives de-asserted winner-take-all signals for

all of said input values. (Clearly, as seen in Figs. 10-12 and Fig. 14, there is an adder tree – the rest of the limitation, specifically the adder tree with winner-take-all signals, is taught in the rejection to claim 1).

References Spiegel and Tsay teach all the limitations above. The specific details for the computer system are cited in parentheses next to that part of the claim. The real focus of the claim, the winner-take-all circuits in the adder trees, is taught in the rejection to claim 1. Obviously the adder tree in Spiegel Fig. 14 is configured to take two input operands and output one operand / result with the modification of the adder element to include the proceeding two multiplexers in Fig. 3 of Tsay, it clearly has "winner-take-all" signaling. Also, it was already established that it would be obvious to modify the multiplexers such that one of the inputs to each of the multiplexers would be tied to zero. (The point of using zeros is so that the other input of the multiplexer is not floating and is a fundamental of the electrical engineering art.) The right multiplexer shown in each of the adder circuits shown in Fig. 3 is tied into both inputs with one being the inverse of the other input, which basically has one input controlling both inputs, so having one tied to zero would be an obvious modification to one of ordinary skill in the art. (The point of using zeros is so that the other input of the multiplexer is not floating and is a fundamental of the electrical engineering art.) This would create the functionality recited above, where the winner-take-all bits are nothing other than control / select bits for the multiplexers (for instance labeled elements 54 and 56 in Fig. 3 Tsay). Specifically, if the right input to each multiplexer were tied (to zero), then a "winner-take-all" signal would merely send the desired operand into one side of the

adder and zero into the other, thus producing the recited input operand. If no winner-take-all signal were asserted, then the adder would function normally and produce a sum of the two input operands as under normal operating conditions. Clearly, again, Tsay is analogous art to Spiegel, as Tsay is directed to a division apparatus (which the selector 90 of Spiegel in Fig. 8 would be utilizing anyway) and the disclosed adder block could obviously be used to form portions of the adder blocks for the adder tree in Spiegel.

It would have been obvious to one having ordinary skill in the art at the time the invention was made to combine the Spiegel adder tree and the adder structure of Tsay, as this would modification would allow the user or system that controlled an adder block within the tree of Spiegel to specify whether or not a branch of the adder tree should be ignored (e.g. passed a zero through the multiplexer)(The point of using zeros is so that the other input of the multiplexer is not floating and is a fundamental of the electrical engineering art) or not.

21. As to claim 28,

The computer system of claim 27, wherein the adder cells generates the sum in response to two or more data valid signals corresponding to the two or more input values, wherein the adder cell includes in the sum those input values whose data valid signals are asserted.

References Spiegel and Tsay teach this limitation. As mentioned above, the winner-take-all signals would merely be control / select signals for the multiplexers. Reference Spiegel clearly teaches the existence of data-valid signals, as in Fig. 8 there

is a "validity bit coder" (element 84) and a "validity decoder" (element 88), which clearly establishes the existence of validity bits. The selector 90 ignores bits output from the scaler 94 unless they have a valid bit asserted. Therefore, it would be obvious to move that functionality from the selector to the adder tree itself, as it would allow the system to discard unused and invalid earlier in the pipeline, which would *prima facie* make the system more efficient and require less processing power. Such validity bit checks could be done by either another multiplexer above the adders (e.g. another one in-line with elements 54 and 56 in Fig. 3 of Tsay) or could be done by using some logic means added to the block.

It would have been obvious to one having ordinary skill in the art at the time the invention was made to combine the Spiegel adder tree and the adder structure of Tsay, as this would modification would allow the user or system that controlled an adder block within the tree of Spiegel to specify whether or not a branch of the adder tree should be ignored (e.g. passed a zero through the multiplexer) or not.

22. As to claim 29,

The computer system of claim 27, further comprising a keyboard device.

Reference Spiegel teaches this limitation. Reference Spiegel clearly teaches the use of a 386-processor based workstation, which is *prima facie* obvious and well known in the art to have a keyboard. A keyboard is inherent to computer-based systems available at the time of filing of Spiegel (e.g. 1996-1997). Since only the primary reference is utilized, no separate motivation or combination is required.

23. As to claim 30,

The computer system of claim 27, further comprising a display device operable to display said pixels.

Reference Spiegel teaches this limitation. Reference Spiegel clearly teaches the use of a 386-processor based workstation, which is *prima facie* obvious and well known in the art to have a monitor. A monitor is inherent to computer-based systems available at the time of filing of Spiegel (e.g. 1996-1997). Further, Spiegel teaches the use of a color VLSI in Fig. 42/2. Figures 46-48 for example are examples of pixels on a monitor. Further, color monitors are disclosed in 3:60-67, 32:62-67, 33:15-37, et cetera. Since only the primary reference is utilized, no separate motivation or combination is required.

Allowable Subject Matter

24. Claims 2-8, 11-14, 18-22, and 24-26 would be allowable if rewritten to overcome the rejection(s) under 35 U.S.C. 112, 2nd paragraph, set forth in this Office action and to include all of the limitations of the base claim and any intervening claims.

The following is a statement of reasons for the indication of allowable subject matter: Adder blocks having the specific traits recited above are not found in the prior art. Specifically, adder blocks containing both multiplexers connected with select bits being passed additionally through another logic gate (an OR) and being passed to the next stage of the adder tree, as well as data valid signals that are configured the same way as the "winner-take-all" signals are not found in the prior art. That is, the combination of multiple control signals (e.g. data valid and winner-take-all) has not been seen or found in the prior art.

Conclusion

26. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure – US PGPub 2002/0059509. At least one patent has been granted on similar subject matter, and clearly as shown it has a WTA/SUM adder tree circuit as recited by applicant.

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Eric V Woods whose telephone number is 703-305-0263. The examiner can normally be reached on M-F 7:30-5:00 alternate Fridays off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Michael Razavi can be reached on 703-305-4713. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Eric Woods

January 13, 2005

JEFFERY BRIER
PRIMARY EXAMINER